

APM466 Assignment 1

Punnawit Payapvattanavong, Student #: 1010071663

February 9, 2026

Fundamental Questions - 25 points

1. In one sentence each:

- (a) Issuing bonds allows governments to raise money without causing inflation as if a government prints too much money, the supply of money may grow faster than the economy's output causing people to lose trust in the currency leading to inflation.
- (b) The long-term part of the yield curve might flatten if investors expect economic growth to slow down, leading to lower inflation and interest rates in the future, reducing the return on long-term bonds.
- (c) Quantitative easing is when the Federal Reserve creates money to buy large amounts of Treasury and mortgage-backed securities, and since the start of the COVID-19 pandemic, the (US) Fed has used Quantitative easing to stabilize financial markets, lower long-term interest rates, and support the economic recovery.

2. In order to construct the 0-5 year yield curve, you would want to select bonds so that each period has at least one bond maturing in that period. This is so that you can guarantee that you can back the spot rate out using the bootstrapping method. You would also want to minimize how much you would need to extrapolate data. With this in mind, I would pick the following bonds:

Can 0.25 Mar 26, Can 1 Sept 26, Can 1.25 Mar 27, Can 2.75 Sept 27, Can 3.5 Mar 28, Can 3.25 Sept 28, Can 4 Mar 29, Can 3.5 Sept 29, Can 2.75 Mar 30, Can 2.75 Sept 30.

Since March and September are exactly 6 months apart, we can quite conveniently extract the spot rates without needing to extrapolate at all.

3. The eigenvectors would tell us how the different processes move together and the associated eigenvalue tells us by how much does the processes move together by.

Empirical Questions - 75 points

4.

(a) See reference for plot.

(b) Pseudocode:

```
for day in days: # Calculate spot curve for each day
    time_to_maturity, spot_rates = [], []
    for bond in day:
        # Loop through each bond
        # Assume bonds already sorted by maturity date
        price, coupon, maturity = bond[price], bond[coupon_payment], bond[maturity]
```

```

time_to_maturity.append(number_of_days(maturity - day.current_day))
for n in number of periods before current period:
    # Subtract the discounted coupon payment from each period
    price -= coupon/(1+spot_rates[n])**time_to_maturity[n]
spot_rates.append(((100+coupon)/price)**1/time_to_maturity[n] - 1)

```

Note that we've assumed annual compounding for spot rates.
See reference for plot.

(c) Pseudocode:

```

forward_dict = {}
for day in days: # Calculate forward rate for each day.
    forward_rates = []
    for bond in range(2, number_of_bonds, 2):
        # Every 2 bonds is 1 year passed.
        t = number_of_days(maturity[bond - 2] - day.current_day)
        n = number_of_days(maturity[bond] - maturity[bond - 2])
        s1 = spot_rate[bond - 2]
        s2 = spot_rate[bond]
        forward_rates.append((((1+s2)**(t+n) / (1+s1)**t)**(1/n)-1)
    forward_dict[day] = forward_rates

```

Note again that we're assuming annual compounding.
See reference for plot.

5. Calculation on github.
6. The eigenvector of the biggest eigenvalue represents how the variation in the bond prices look whilst the eigenvalue tells us how much of the variation in bond price moves according to the eigenvector.

References and GitHub Link to Code

1. Canadian Bond data
2. Github link
3. Information on Canadian government bonds



